

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERÍA EN INFORMÁTICA

SIMOCAP, Sistema Simple de Captura de Movimiento

Realizado por
JOSÉ ALBERTO GANDULLO ÁVILA

Dirigido por
JOSÉ CORTÉS PAREJO

Departamento
MATEMÁTICA APLICADA I

Sevilla, Junio de 2010

Agradecimientos a toda mi “familia”
en el sentido más amplio de la palabra.

Y al III Premio de Sevilla adscrito al IV Concurso Universitario de Software Libre
que me concedió una Mención Especial por este proyecto.

Blog del proyecto:
http://simocap.wordpress.com
Forja de Desarrollo:
https://forja.rediris.es/projects/cusl4-simocap
Correo de contacto:
yoalberto007@gmail.com

9. Manual de Usuario

9.1 Sistemas Operativos soportados

Hasta el momento el código solo ha sido probado en plataforma linux pero teniendo en cuenta que su principal dependencia que es ARToolKit es multiplataforma (Windows, Linux y Mac) y el código está escrito en lenguaje C, es de esperar que no entrañe muchas dificultades el migrarlo a otras plataformas.

9.2 Instalación y Uso

9.2.1 ArToolkit

Nota: Solo se indican las instrucciones para instalación en una plataforma linux

Prerrequisitos:

Instalar los siguientes paquetes o la versión actual más parecida desde Synaptic :

freeglut3-dev
libgstreamer0.10-dev
libgstreamer-plugins-base0.10-dev (quizas no necesario)
libxi-dev
libxmu-headers
libxmu-dev
libjpeg62-dev
libglib2.0-dev
libgtk2.0-dev

Montando (compilando) ARToolKit

Ahora se puede montar ARToolKit. Abrir un terminal de consola.

Descomprimir el archivo de ARToolKit en una localización que convenga. La ruta de esa localización será referida a partir de este momento como {ARToolKit}:

```
tar zxvf ARToolKit-2.72.1.tgz
```

Configuración y compilado.

Moverse a la carpeta donde hemos descomprimido ARToolKit y teclear `./Configure.`

```
cd {ARToolKit}
./Configure
```

Aparecerá un menú con opciones, para el caso de este proyecto seleccionar la opción *5 Gstreamer Media Framework.*

Una vez terminado el proceso de configuración ejecutar el comando make para compilar el código.

```
make
```

Si el proceso de compilado ha terminado con éxito se puede considerar que ARToolKit ya está instalado.

9.2.2 Simocap

Instalación

Para instalar Simocap se deben seguir los siguientes pasos.

Descomprimir el archivo s*imocap_source.zip* en la carpeta examples de ARToolKit de forma que se cree la carpeta simocap_source**.**

Abrir un terminal de comandos dentro de la carpeta simocap_source y ejecutar el comando make

```
make
```

de este modo el programa estará compilado y listo para ejecutar.

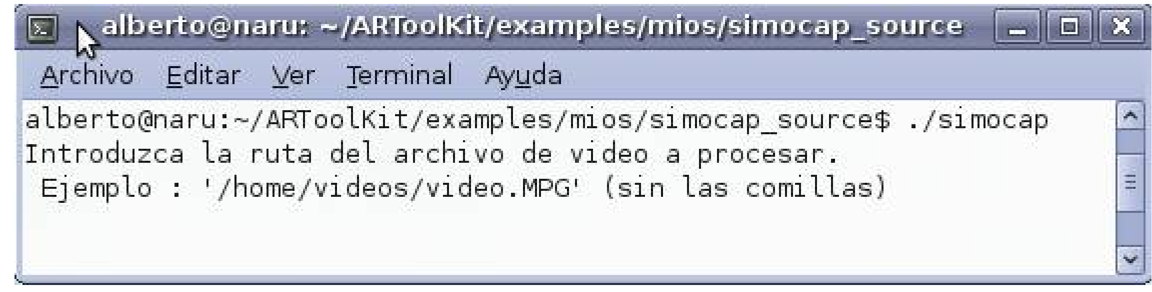
Uso de la aplicación

Para la ejecución de la aplicación debemos tener el vídeo con nuestros movimientos y las marcas colocadas correctamente, Una vez localizado el vídeo abrimos un terminal de consola y nos movemos hasta la carpeta donde se encuentra el ejecutable simocap (carpeta simocap_source).

Tecleamos "./simocap" para ejecutar,

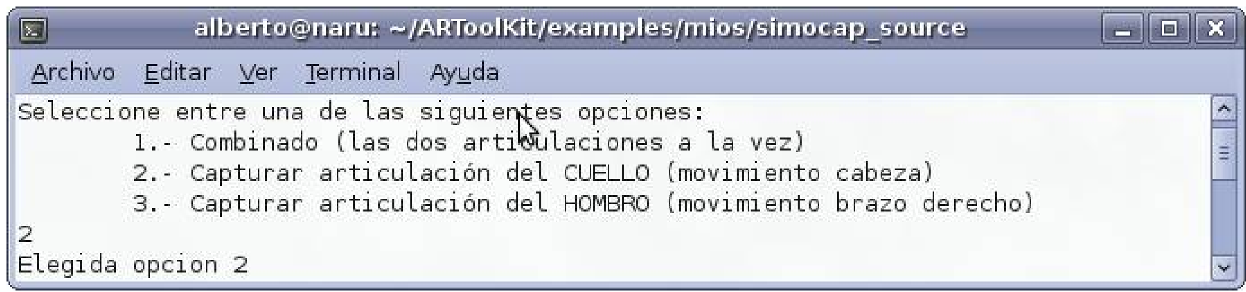
```
./simocap
```

inmediatamente nos pedirá la ruta de localización del vídeo a analizar



Ejecución de simocap (función de localización)

Una vez se teclea la ruta, se ofrece un menú con varias opciones de captura según se quiera capturar los movimientos de unas partes del cuerpo u otras.

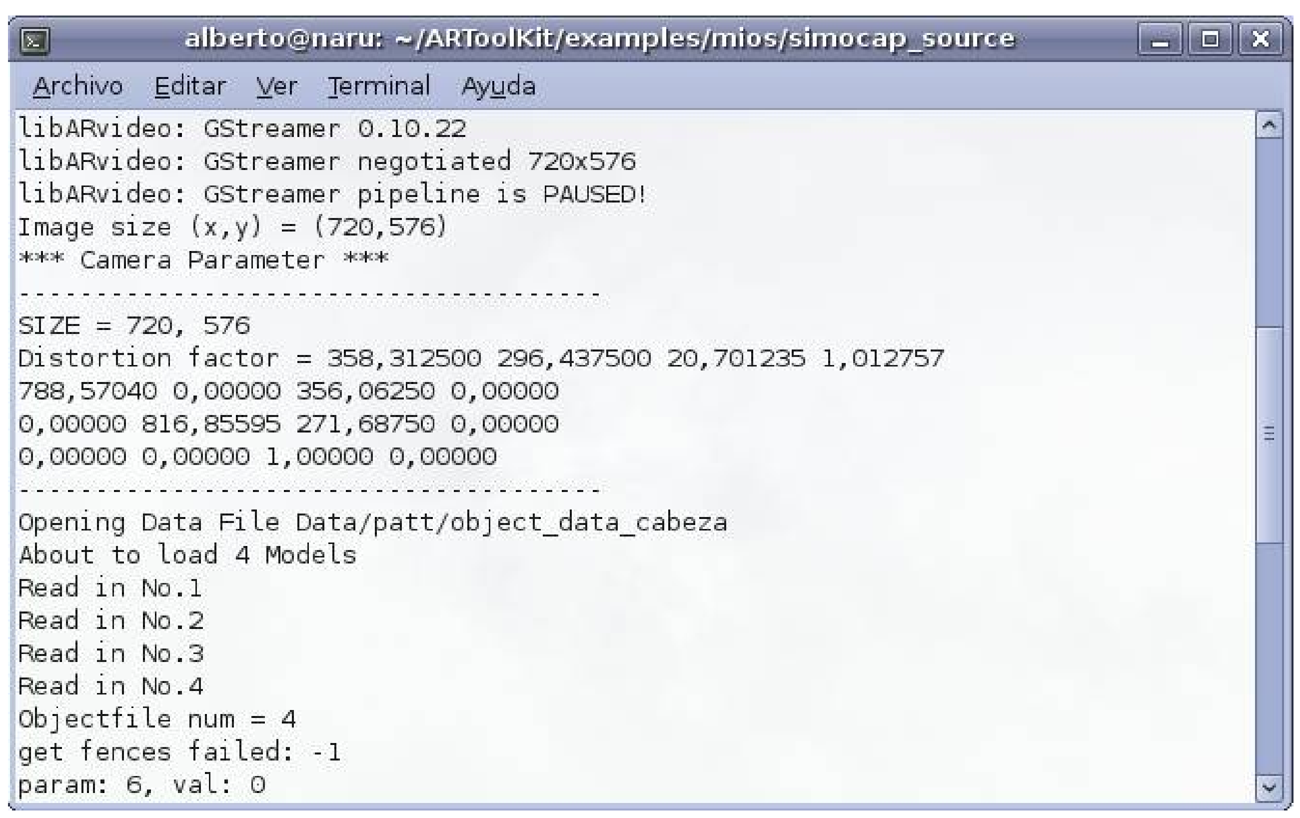


Ejecución de simocap (función captureOption)

Al seleccionar una de las opciones empezará el análisis , (si no hay error en la ruta) , lo podremos ver ya que se abrirá una ventana en la que se reproduce el vídeo.

Para comenzar la captura del movimiento al archivo bvh tenemos que pulsar ESPACIO.

Las marcas capturadas serán superpuestas con una figura en 3 dimensiones (realidad aumentada). Además en la consola se irá imprimiendo una serie de datos referentes al proceso



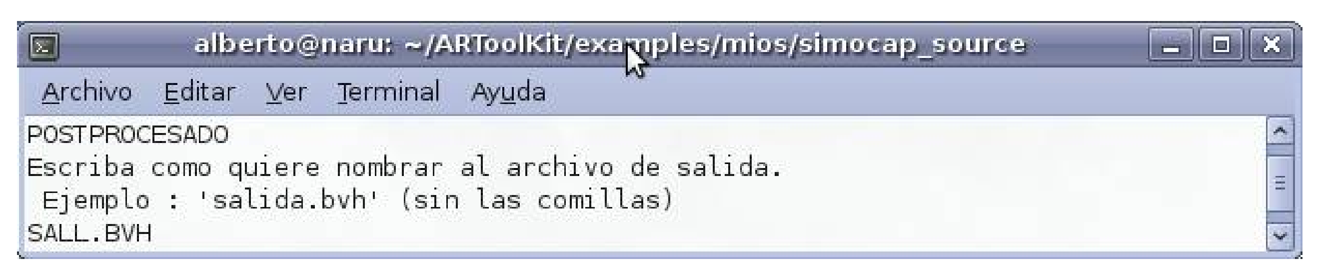
Ejecución de simocap (función init)

Para detener la captura de movimiento pulsaremos ESC ,(no cerrar ventana de reproducción con X porque el proceso no terminará correctamente)



Ejemplo de ventana de reproducción de vídeo

Al pulsar Esc la ventana de reproducción se cerrará, y debemos volver a consola, ya que allí nos está pidiendo que insertemos por teclado el nombre del archivo de salida, mejor que termine en .bvh, así no tendremos problemas al reproducirlo.



Ejecución de simocap (función postprocesado)

Entonces terminará el proceso ya tenemos en nuestra carpeta de simocap o en la ruta que hayamos indicado con el nombre de nuestro archivo bvh de salida, ahora lo podemos reproducir normalmente con qavimator o cualquier otro programa para este fin.

El código fuente de SIMOCAP se puede encontrar en el CD-ROM adjunto a esta memoria.

9.2.3 Otros

Para la visualización de los archivos de salida que produce la aplicación Simocap, será necesario instalar software que provea este servicio.

Ejemplos de software que usa archivos en formato BVH:

- Lightwave 3D
- 3ds Max
- Blender
- Cobalt
- Daz Studio
- Esenthel Engine
- Lifeforms
- Maya
- Poser
- Seamless3d Free open source modeller
- Second Life
- Avimator y su Qt port Qavimator
- Maxon Cinema 4d
- formZ
- Milkshape 3D
- Papervision 3D

Para la realización de pruebas y visualización de los archivos BVH de salida durante el desarrollo de Simocap se usó la aplicación Qavimator por su sencillez y por ser software libre.

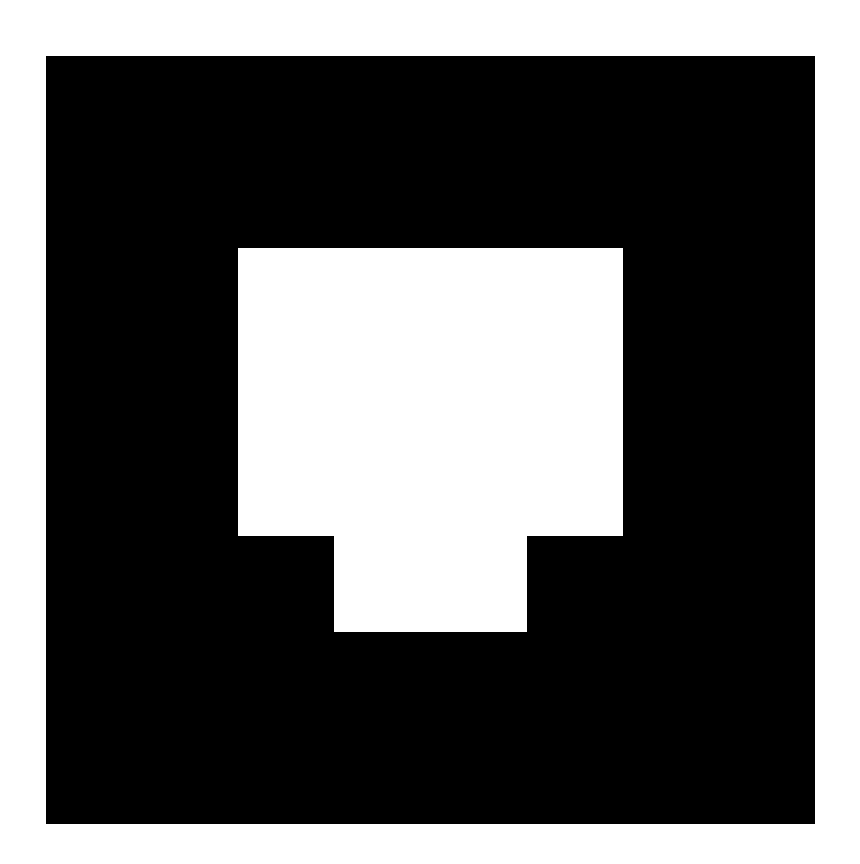
Para la instalación de ésta u otras aplicaciones de la lista anterior consultar sus respectivas webs.

9.3 Construcción de dispositivos de seguimiento

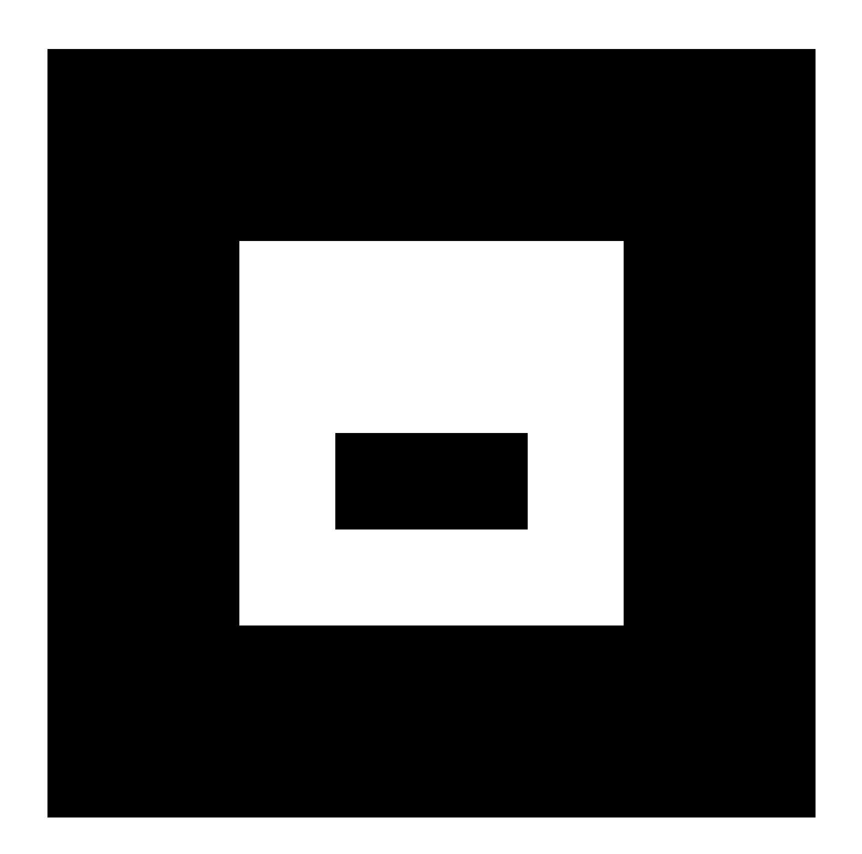
Marcadores para impresión

A continuación se muestran los patrones que habría que imprimir para colocarlos en los dispositivos de seguimiento indicados. Al imprimirlos conviene dejar espacio en blanco suficiente alrededor de la cuadricula negra exterior, para que los algoritmos de visión por computador capten el el contraste, por ello en este documento se ha colocado un patrón en cada página y así poder obtener los marcadores desde este documento directamente.

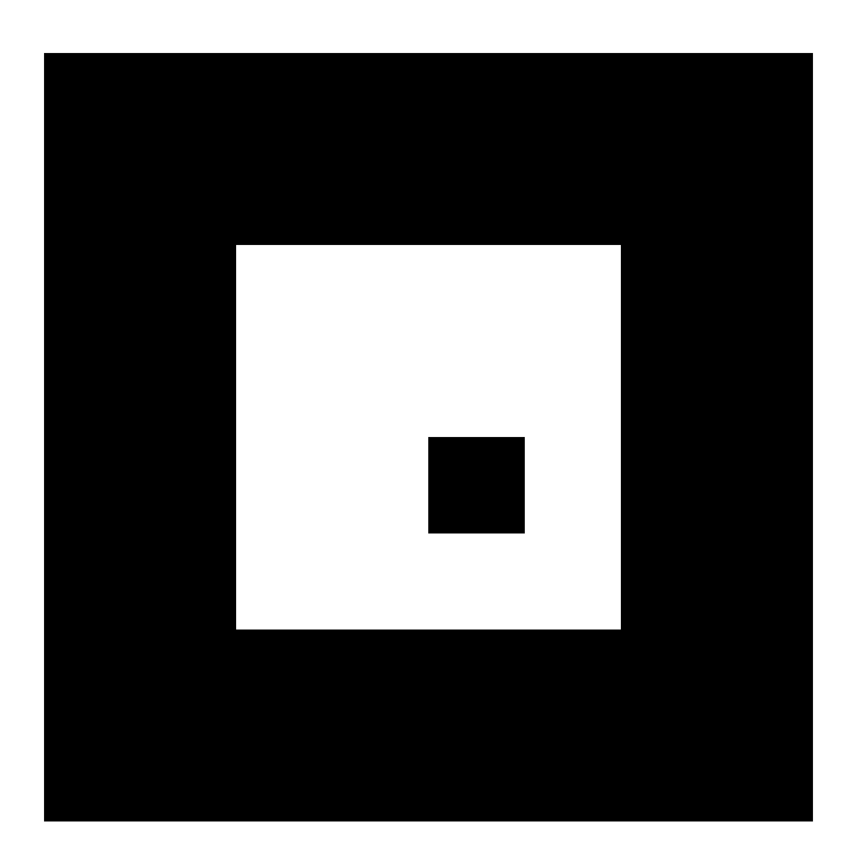
Marcador base, para el dispositivo del pecho:



Marcador frontal para el dispositivo de la cabeza:



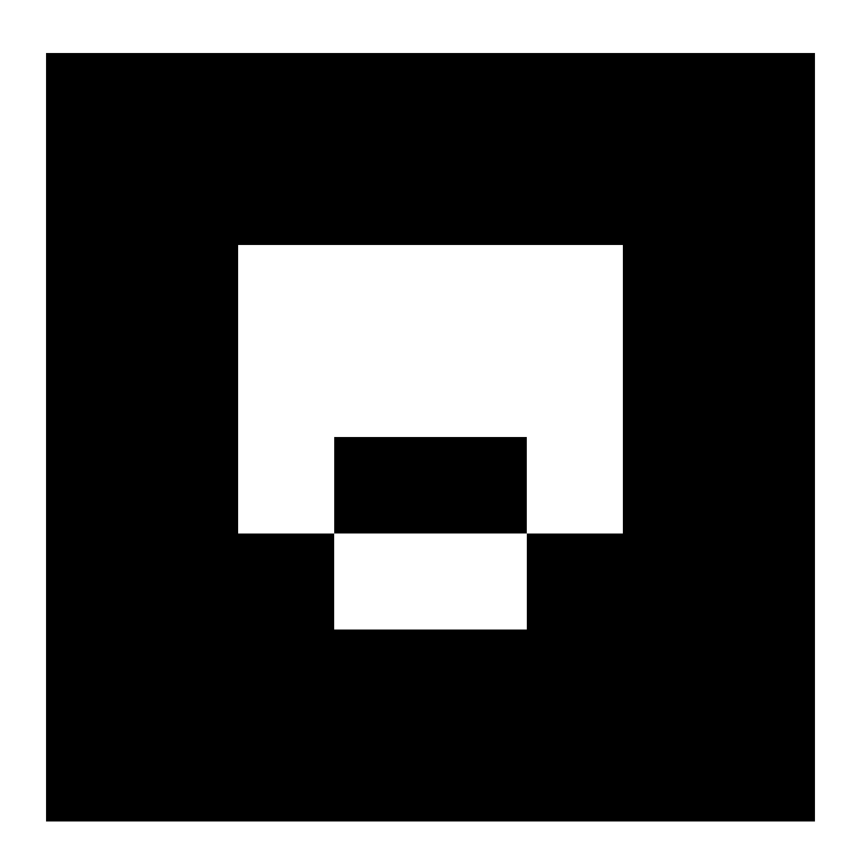
Marcador lateral izquierdo para el dispositivo de la cabeza:



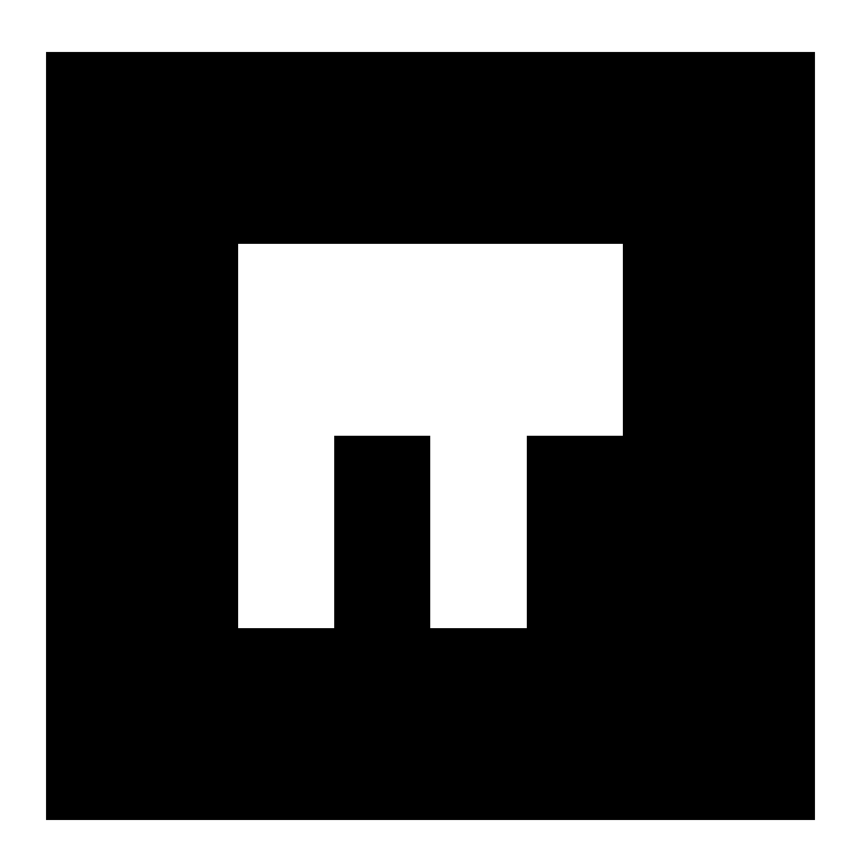
Marcador <u>lateral derecho</u> para el dispositivo de la cabeza:



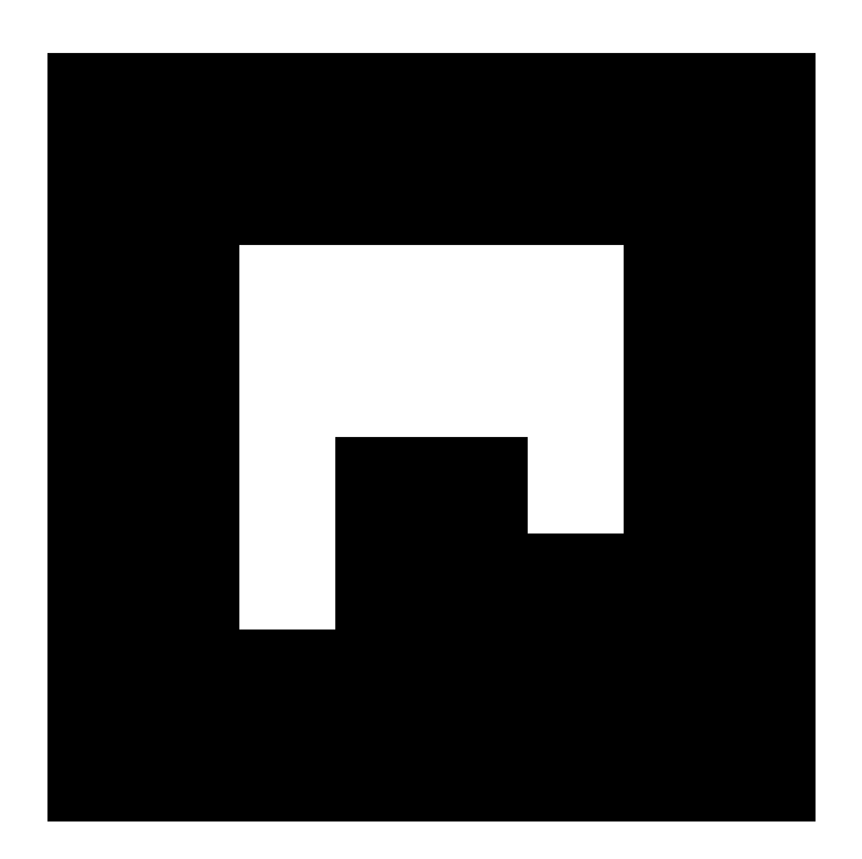
Marcador frontal para el dispositivo del brazo:



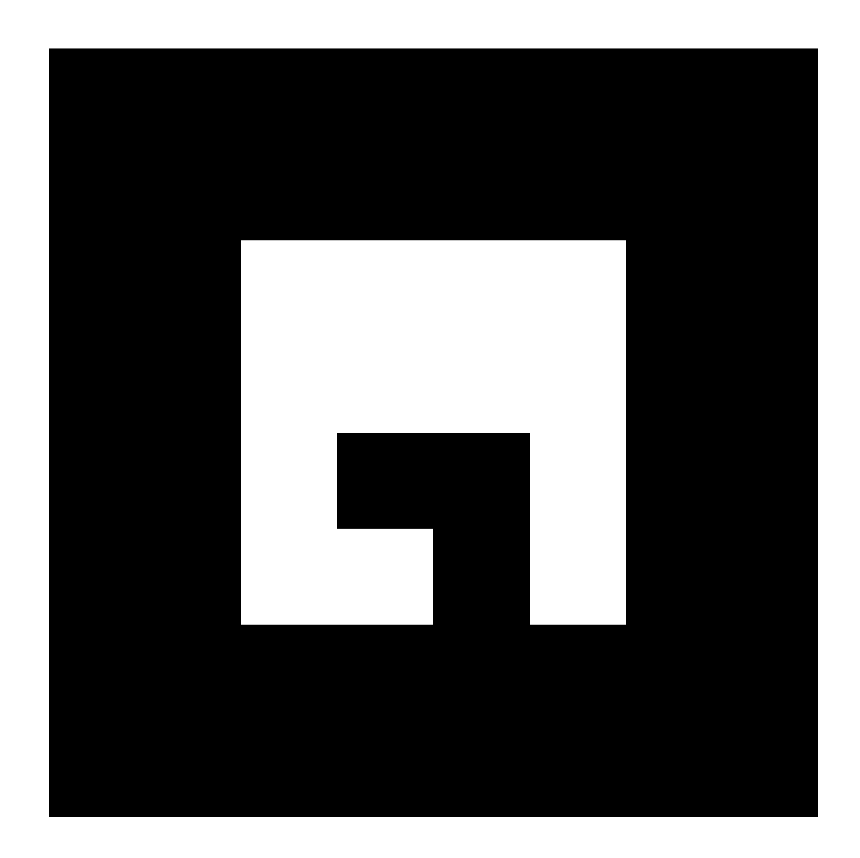
Marcador inferior para el dispositivo del brazo:



Marcador superior para el dispositivo del brazo:



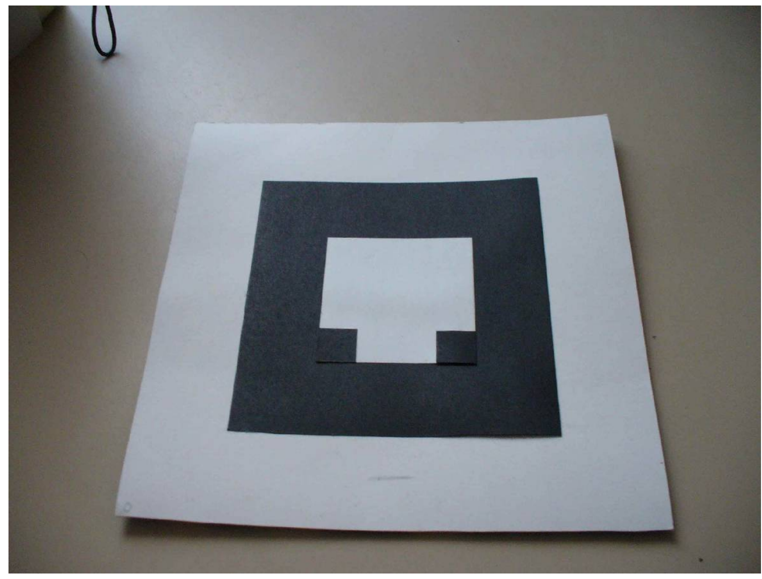
Marcador posterior para el dispositivo del brazo:



Construcción de los dispositivos:

Dispositivo base, para el pecho.

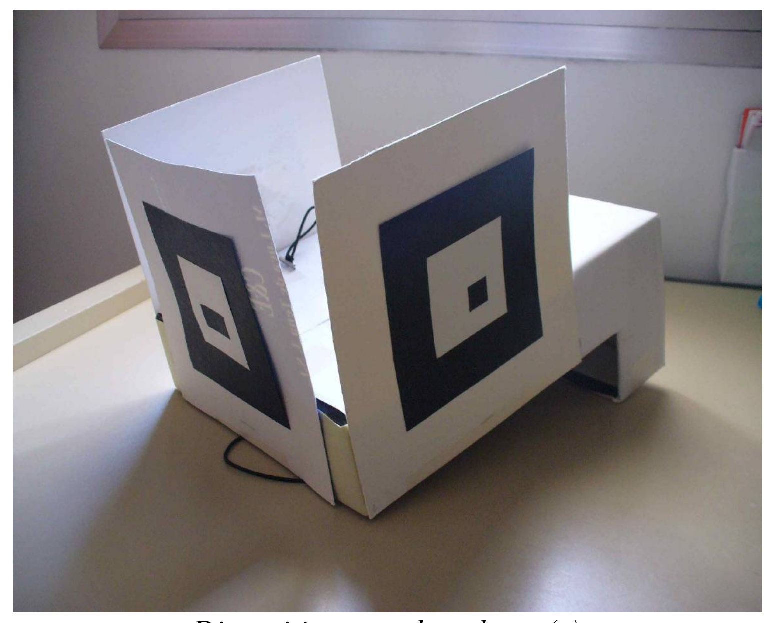
Es el más simple y consiste únicamente en una superficie plana, a la cual se le pone el marcador correspondiente de forma visible y algún tipo de adhesivo o sistema de sujeción para colocarlo en el pecho.



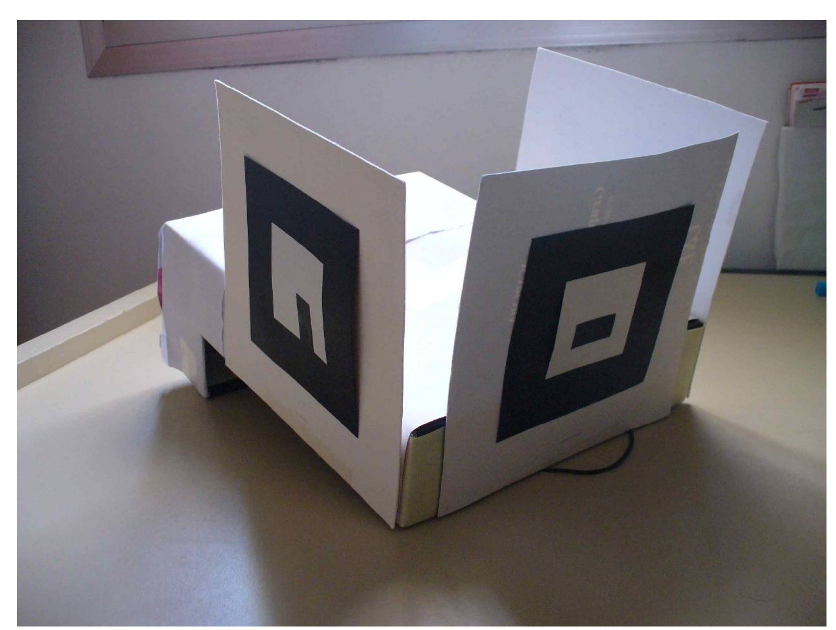
Dispositivo base para el pecho

Dispositivo para la cabeza

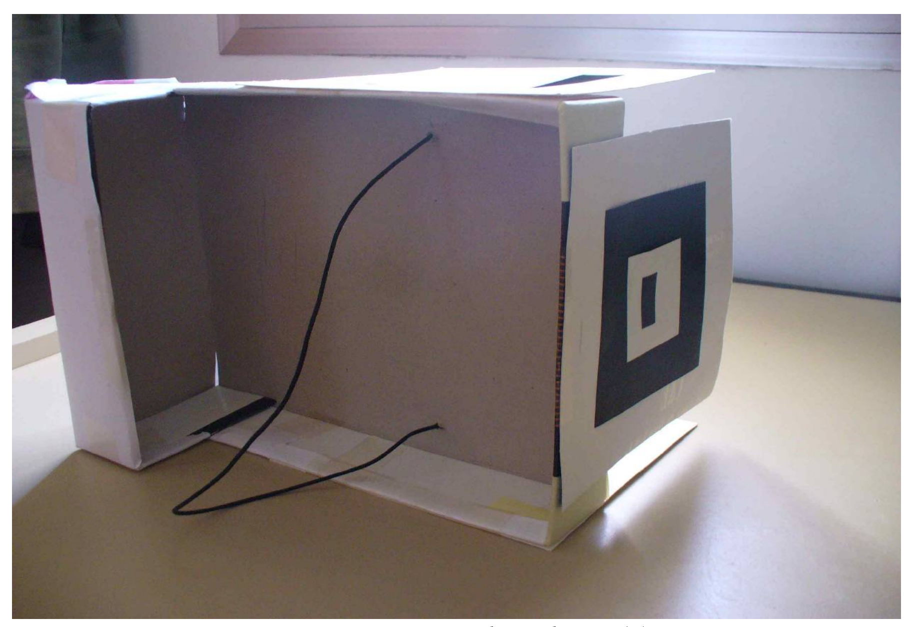
El dispositivo para la cabeza es una especie de casco que usa para su sujeción una goma elástica. La base está fabricada con la tapadera de una caja de zapatos que se dobla en la parte posterior para mejorar el agarre, en los bordes de la misma se adhieren los marcadores, previamente estos se han pegado a una cartulina o cartón para mejorar su rigidez.



Dispositivo para la cabeza (a)



Dispositivo para la cabeza (b)



Dispositivo para la cabeza (c)

Dispositivo para las extremidades o cubo

Este dispositivo es algo más complejo en su fabricación, pero también su calidad y durabilidad es mayor, además su diseño hace que sea posible su uso en otras partes del cuerpo y no solo en el brazo (antebrazo, mano, pierna).



Dispositivo para articulaciones (cubo)

Pasos para la construcción:

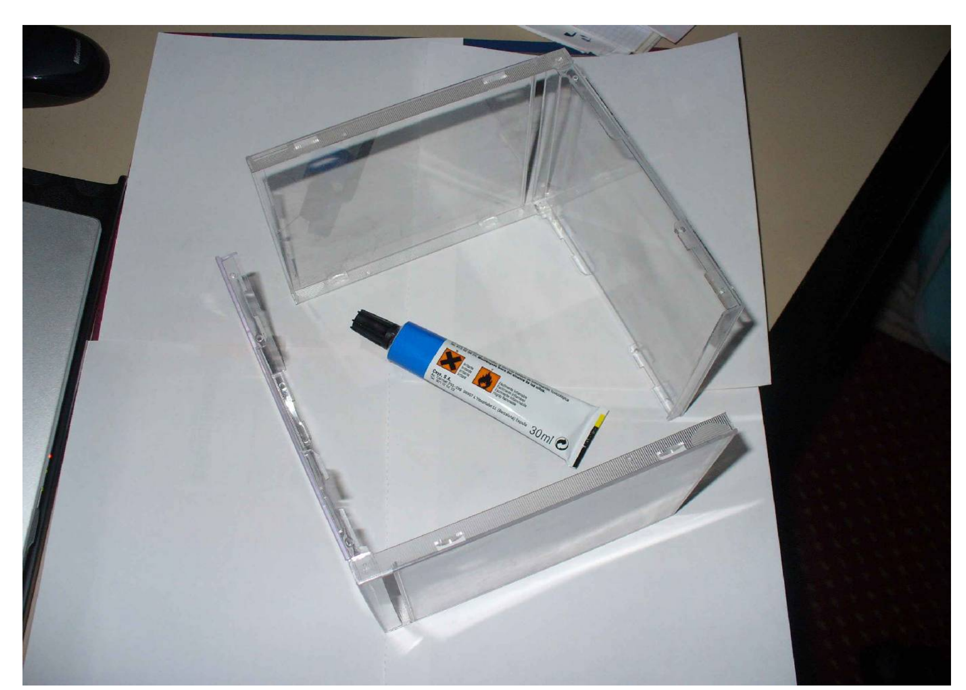
Se necesitan 2 carátulas (cajas) de cd normales.

Se le quita el plástico negro interior a las carátulas y se quedándose como la de abajo,



Desmontar las carátulas

Se colocan las carátulas en posición y se aplica pegamento en las esquinas tanto en las de contacto como en las de los ejes de rotación.



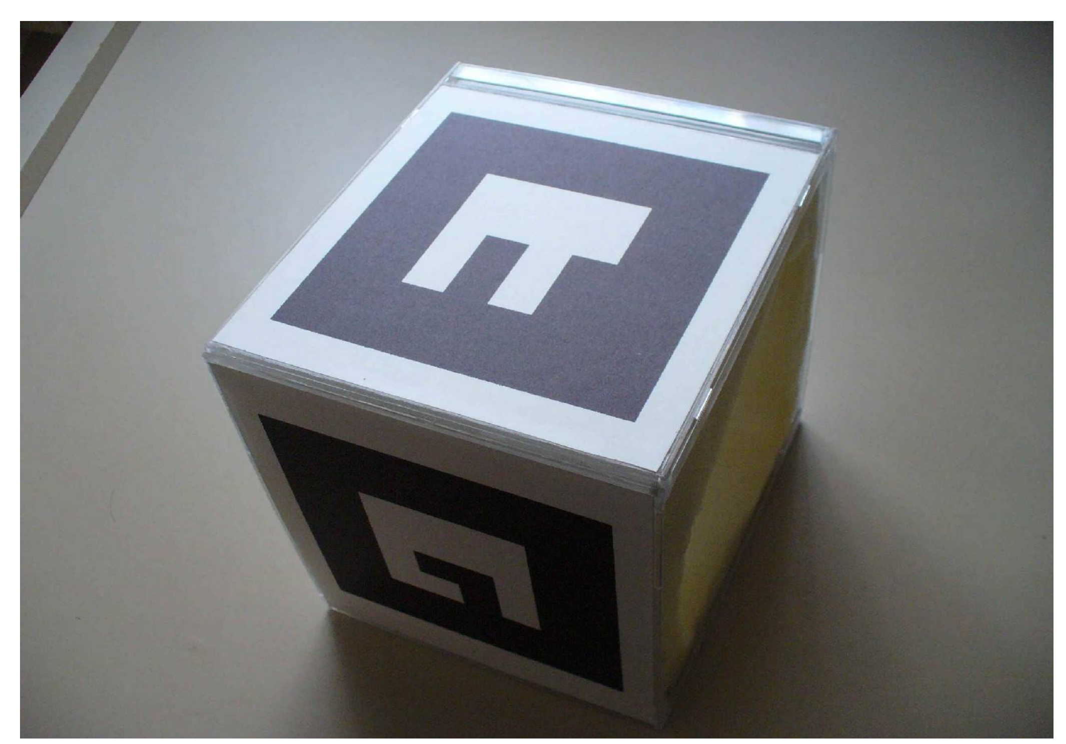
Aplicar pegamento para fijar la estructura

Se deja secar, obteniendo el siguiente resultado



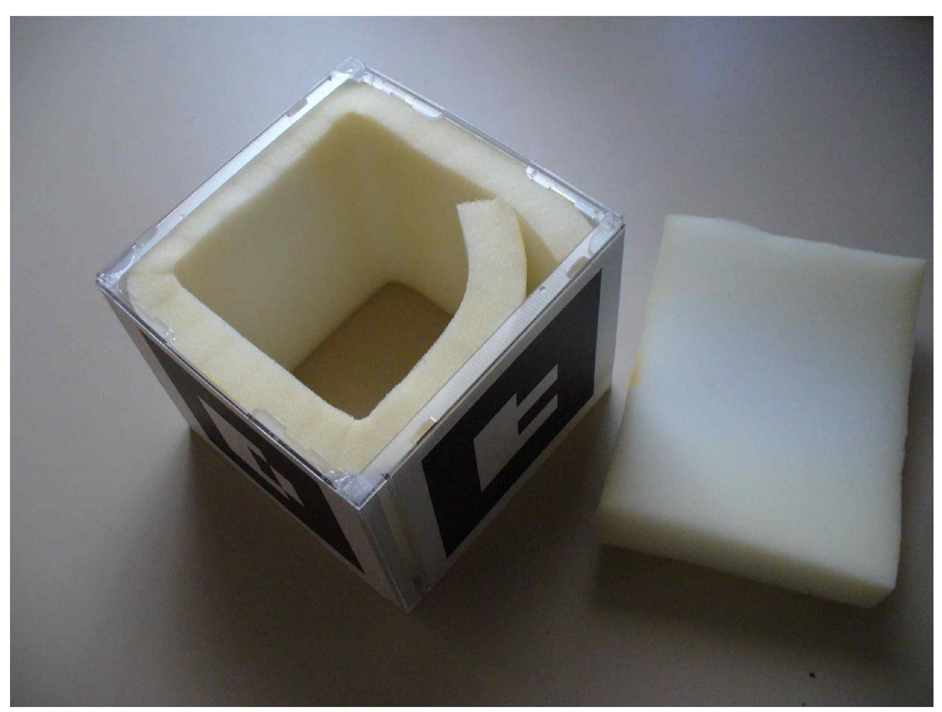
Estructura del dispositivo

Ahora se imprimen los patrones correspondientes y se colocan en las caras adecuadas



Estructura del dispositivo con marcadores adheridos

Para finalizar se le introduce un trozo de poliespan para que se ajuste al brazo, o extremidad correspondiente.



Añadir poliespan para mejorar la sujeción

Y ya tenemos nuestro dispositivo para el brazo.

10. Apendice

10.1 Herramientas usadas para el desarrollo del proyecto

Durante el desarrollo de este proyecto se ha procurado que todo el software implicado fuera software libre. Esto se debe a motivaciones principalmente éticas entendiéndose que el software es conocimiento y debe poderse difundir sin trabas. Su ocultación es una actitud antisocial y la posibilidad de modificar programas es una forma de libertad de expresión siempre protegiendo el derecho del autor ha ser reconocido como tal.

Sin más, a continuación se hace una descripción de las herramientas usadas.

Ubuntu 9.04
Sistema operativo distribución de Linux orientada a escritorio, basada en Debian GNU/Linux pero enfocado en la usabilidad, lanzamientos regulares y fácil de instalación

Terminal , linea de comandos
Usado para la instalación, compilación y ejecución de programas.

Editor de Eclipse
Eclipse es un entorno de desarrollo software muy avanzado y multilenguaje pero en este proyecto solo se ha usa su editor para programar el código en un entorno más amigable, la compilación y generación de ejecutables se ha hecho con Make.

Make, makefile
Make es una herramienta de generación o automatización de código, muy usada en los sistemas operativos tipo Unix/Linux. Por defecto lee las instrucciones para generar el programa u otra acción del fichero makefile. Las instrucciones escritas en este fichero se llaman dependencias.
La herramienta make se usa para las labores de creación de fichero ejecutable o programa, para su instalación, la limpieza de los archivos temporales en la creación del fichero..., todo ello especificando unos parámetros iniciales (que deben estar en el makefile) al ejecutarlo.

Forja de Rediris
Una forja es una plataforma de desarrollo colaborativo de software. Se enfoca hacia la cooperación entre desarrolladores para la difusión de software y el soporte al usuario.
En este tipo de plataformas se albergan múltiples proyectos de software, en los que los desarrolladores han de registrarse para poder contribuir. Consta de numerosas aplicaciones normalmente con interfaz web para la administración y desarrollo de estos proyectos en común.

La pagina del proyecto en la forja de Rediris es la siguiente:

https://forja.rediris.es/projects/cusl4-simocap/

RapidSVN, para Sistema de control de Revisiones
El Sistema de Control de Revisiones (Revisión Control System - RCS) es una implementación de software para el control de versiones que automatiza las tareas de guardar, recuperar, registrar, identificar y mezclar versiones de archivos. RCS es útil para archivos que son modificados frecuentemente, por ejemplo programas informáticos, documentación, gráficos de procedimientos, monografías y cartas.

Gtk-RecordMyDesktop

Es un programa que nos permite grabar nuestro escritorio en vídeo. Mediante esta aplicación podremos captar imágenes de cualquier interacción con los menús del sistema, acciones secundarias, así como otros programas que ejecutemos.

Gimp

Es un programa de edición de imágenes digitales en forma de mapa de bits, tanto dibujos como fotografías. Es un programa libre y gratuito. Está englobado en el proyecto GNU y disponible bajo la Licencia pública general de GNU.

Blog **http://simocap.wordpress.com/**

Usado para facilitar el seguimiento del avance del proyecto y para compartir los conocimientos adquiridos durante el transcurso del Concurso Universitario de Software Libre. La afluencia de visitas al blog está en una media de 10 a 20 visitas diarias Y si en google se hace una búsqueda por la palabra "simocap" este blog aparece el primero de la lista.

Qavimator

QAvimator es una herramienta simple para crear y editar animaciones de un personaje virtual en este proyecto se usará para el estudio y visualización de los archivos BVH que se generan como salida.

OpenOffice

Es una suite ofimática libre (código abierto y distribución gratuita) que incluye herramientas como procesador de textos, hoja de cálculo, presentaciones, herramientas para el dibujo vectorial y base de datos.

Cámara de vídeo sony handycam

Usada para grabar los vídeos del actor para capturar sus movimientos.



Cámara de vídeo Sony Handycam

10.2 Análisis Temporal

El planteamiento del proyecto se produce el 15 de Septiembre de 2009 con la apertura del plazo de inscripción al IV Concurso Universitario de Software Libre. Durante el periodo de tiempo transcurrido entre la apertura del plazo y la confirmación de participación fechada a 1 de Diciembre, el desarrollo del proyecto se limitó a las fases de investigación y diseño de objetivos.

Computando una media de 7 horas semanales durante las 10 semanas de este primer periodo hacen un total de 70 horas.

Una vez confirmada la participación en el concurso, se comienza el desarrollo más intenso del mismo y la fase de programación de pruebas para comprobar las posibilidades de las tecnologías investigadas. Esta fase dura hasta Febrero de 2010.

Computando una media de 14 horas semanales durante estas 9 semanas hacen un total de 126 horas.

En febrero se anunció la fecha de la final del Premio de Sevilla adscrito al Concurso Universitario de software libre, sería el 16 de abril, eso hizo incrementar el ritmo de trabajo y empezar a programar la aplicación real, diseñar y construir los dispositivos y documentar los avances en el blog.

Computando una media de 20 horas semanales durante las 11 semanas de este periodo hacen un total de 220 horas,

Finalmente después de la entrega de premios en el premio local de Sevilla del concurso en el cual fui finalista y recibí una mención especial, empezó la fase de mejoras y de redacción de la memoria final para el proyecto fin de carrera, periodo que duró hasta la primera semana de Junio cuando se entregó esta memoria.

Computando una media de 20 horas semanales durante las 7 semanas de este último periodo contabilizado hacen un total de 140 horas.

El tiempo total en el que se estima que se ha desarrollado el proyecto es de unas 556 horas.

10.3 Análisis de Costes

El coste se puede resumir en las siguientes tres partes.

Coste del Software utilizado.

Como ya se ha señalado en puntos anteriores se ha usado software libre para el desarrollo y aunque el software libre no es gratuito por definición en el caso de las herramientas usadas para este proyecto si lo han sido. Por lo tanto en software el coste ha sido de 0 euros.

Coste de construcción de los dispositivos de seguimiento.

La mayoría de los materiales usados para la construcción de estos dispositivos son materiales reciclados a los que se les da un segundo uso distinto al de fabricación. A pesar de esto hay otros materiales que si han tenido un coste aunque pequeño. Entre estos materiales estarían el pegamento, el poliespan, y la tinta de impresora para los marcadores. Podemos estimar el coste de estos materiales en unos 10 euros.

Coste de la Cámara de Vídeo



Cámara de vídeo Sony Handycam

Decir que la cámara de vídeo usada durante el transcurso de este proyecto es prestada, pero en mercado tiene un precio aproximado a 200 euros.

Por lo tanto el coste total del proyecto asciende a unos 210 euros. Comparando este coste al precio de los sistemas habituales de captura de movimiento el cual suele rondar entre los 30 mil y los 100 mil euros, se puede calificar el coste del sistema Simocap de insignificante.

11. Bibliografía

Publicaciones

Manuel Escribano Cauqui
Programación de Graficos en 3D
RA-MA (1995)
84-7897-156-4

Parent, Rick; Ebert, David S.
Computer Animation Complete All-in-One
Morgan Kaufmann Publishers (Elsevier) (2010)
978-0-12-375078

Magnenat-Thalmann, N. & Thalman, D.
Synthetic Actors in Computer-Generated 3D Films
Springer-Verlag (1990)
0-387-52214-X

Otras referencias

Cortés, J.
Cambio entre Sistemas de Referencia
Cinemática de la cadena Hombro-Codo-Muñeca. Representación mediante ángulos de Euler RPY
Departamento de Matemática Aplicada I, Universidad de Sevilla (2008)

Kato, H. & Billinghurst, M.
Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System
Faculty of Information Sciences, Hiroshima City University
Human Interface Technology Laboratory, University of Washington (1999)

E. Woods, P. Mason, M. Billinghurst.
MagicMouse: an Inexpensive 6-Degree-of-Freedom Mouse.
Proceedings of Graphite (2003)

Hors Fraile, S. & Cortés, J
Cinemática del Movimiento en Estructuras Articuladas
Departamento de Matemática Aplicada I, Universidad de Sevilla (2008)

Alonso Sanchez, F. J.
Tratamiento de Señales Cinemáticas y de Masas de Tejido Blando en el Ánalisis Dinámico Inverso de Modelos Biomecánicos Esqueleterales
Departamento de Tecnología Electrónica y Electromecánica, Univ. de Extremadura (2005)

Bleser , G.
Towards Visual-Inertial SLAM for Mobile Augmented Reality
Vom Fachbereich Informatik der Technischen Universität Kaiserslautern (2009)

Lander, J.
Working with Motion Capture File Formats
Game Developer (1998)

Gonzalez-Morcillo, C.; Jiménez, L.; Moreno-García, J.
Aplicación de Lógica Difusa en Sistemas de Captura de Movimiento
Universidad de Castilla-La Mancha (2003)

Recursos Electrónicos

Cinemática inversa
<http://es.wikipedia.org/wiki/Cinematica_inversa>

Human Mocaps Xsens
<http://www.xsens.com/en/company-pages/company/human-mocap>

Como crear un Proyecto Libre (2004)
<http://www.somoslibres.org/modules.php?name=News&file=article&sid=256>

Artoolkit
<http://www.hitl.washington.edu/artoolkit/>

Gimbal lock
<http://en.wikipedia.org/wiki/Gimbal_lock>

Euler angles
<http://www.euclideanspace.com/maths/geometry/rotations/euler/index.htm>

List of motion and gesture file formats
<http://en.wikipedia.org/wiki/List_of_motion_and_gesture_file_formats>

The Matrix and Quartenion Faq
<http://www.j3d.org/matrix_faq/>

Biovision BVH
<http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html>

PFC Alberto Corrales García Jugando con las coordenadas de ArtoolkitPlus
<http://pfc-albertocorralesgarcia.blogspot.com/2008/07/jugando-con-las-coordenadas-de.html>